

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-m2-php-problems/grade/jhr4>

IT202-008-S2024 - [IT202] M2 PHP Problems

Submissions:

Submission Selection

1 Submission [active] 2/1/2024 1:38:44 PM

Instructions

^ COLLAPSE ^

Guide:

- 1 .Make sure you're in the main branch locally and ``git pull origin main`` any pending changes
- 2 .Make a new branch per the recommended branch name below (`git checkout -b ...`)
- 3 .Grab the template code from <https://gist.github.com/MattToegel/48b48377eaa1937c886b7840c449750a>
- 4 .Create individual PHP files for each problem and save the files inside your `public_html` folder in a subfolder of your choice
 - 1 .If you don't have this folder yet, refer to the setup lessons (you'll need a few files for the deployment to work)
- 5 .Move the unedited template files to github
 - 1 ``git add .``
 - 2 ``git commit -m "adding template files"``
 - 3 ``git push origin <homework branch>`` (see below and don't include the `< >`)
 - 4 .Create and open a pull request from the homework branch to main (leave it open until later steps)
- 6 .Note: As you work, it's recommended to add/commit at least after each solution is done (i.e., 3+ times in this case)
 - 1 .Make sure the files are saved before doing this
- 7 .Fill in the items in the worksheet below (save as often as necessary)
- 8 .Once finished, export the worksheet
- 9 .Add the output file to any location of your choice in your repository folder (i.e., a `Module2` folder)
- 10 Check that git sees it via ``git status``
- 11 If everything is good, continue to submit
 - 1 .Track the file(s) via ``git add``
 - 2 .Commit the changes via ``git commit`` (don't forget the commit message)
 - 3 .Push the changes to GitHub via ``git push`` (don't forget to refer to the proper branch)
 - 4 .Create a pull request from the homework related branch to main (i.e., `main <- "homework branch"`)
 - 5 .Open and complete the merge of the pull request (it should turn purple)
 - 6 .Locally checkout main and pull the latest changes (to prepare for future work)
- 12 Take the same output file and upload it to Canvas
 - 1 .*This step is new since GitHub renders the PDF as an image the links aren't clickable so this method works better
 - 2 .*Remember, the github process of these files are encouragement for your tracking of your progress

Problem 1 (3 pts.)

^ COLLAPSE ^

Task #1 - Points: 1

Text: Screenshot of the Problem 1 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have only the odd values output.
Requires at least 2 screenshots (code + output from heroku dev)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Edits were done only in the processArray() function and original template code/comments remain untouched
<input checked="" type="checkbox"/> #2	1	Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)
<input checked="" type="checkbox"/> #3	5	Only odd values output (not odd indexes/keys)
<input checked="" type="checkbox"/> #4	1	Includes code comments with student's ucid and date
<input checked="" type="checkbox"/> #5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

☒ Large Gallery


Checklist Items (3)

#1 Edits were done only in the processArray() function and original template code/comments remain untouched

#2 Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)

#4 Includes code



Checklist Items (2)

#3 Only odd values output (not odd indexes/keys)

#5 Output of code includes heroku dev URL with student's ucid visible

Heroku deploy of PHP problem 1

#4 includes code
comments with
student's ucid and
date

Code for PHP problem 1.

Task #2 - Points: 1

Text: Explain your solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Clearly explains how the code/logic solves the problem (mentions how the odd values are determined)

Response:

In this problem I used a while loop that would iterate over each item in the array \$arr by finding it's length by count() and then incrementing it by 1. It was originally a for loop for(\$i=0; \$i<count(\$arr); \$i++), but it wasn't working for some reason when I first tried. Then I used % division to check if the number was odd or even as even numbers would have no remainders while odd numbers would have a remainder of 1. If it had a remainder of 1 (meaning its odd) it would echo/print the current number by using the index number.

Task #3 - Points: 1

Text: Link to Problem 1 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.

Note: This link won't be active until you merge the pull request, so it's expected to not exist at the time of filling this task

URL #1

<https://jhr4-it202-008-prod-af7c79552123.herokuapp.com/PHP-Problems/problem1.php>

Problem 2 (3 pts.)

Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have the numerical values summed and rounded to two decimals (similar to currency)
Requires at least 2 screenshots (code + output from heroku dev)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Edits were done only in the getTotal() function and original template code/comments remain untouched (unless noted)
<input type="checkbox"/> #2	1	Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)
<input type="checkbox"/> #3	5	Passed in array's values get summed AND rounded to two decimal places like currency (i.e., 0.00, 0.10, 1.10)
<input type="checkbox"/> #4	1	Includes code comments with student's ucid and date
<input type="checkbox"/> #5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

☐ Large Gallery



Checklist Items (3)

#1 Edits were done only in the getTotal() function and original template code/comments remain untouched (unless noted)

#2 Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)

#4 Includes code comments with student's ucid and date



Checklist Items (2)

#3 Passed in array's values get summed AND rounded to two decimal places like currency (i.e., 0.00, 0.10, 1.10)

#5 Output of code includes heroku dev URL with student's ucid visible

Heroku deploy of PHP problem 2.

Code for PHP Problem 2.



^ COLLAPSE ^

Task #2 - Points: 1

Text: Explain your solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
---	--------	---------

#	Points	Details
#1	1	Clearly explains how the code/logic solves the problem (mentions both how the values get summed and how the rounding is solved correctly)

Response:

For this I used the same loop as the previous problem, for the same reason to iterate over each item in the array. Then I declared/initialized a \$sum variable outside of the loop as 0. Then for each iteration the sum would be incremented by the current number given from the index (i) of the array. The line "\$sum += \$arr[\$i]" basically means that the \$sum is now assigned to "\$sum + \$arr[\$i]" which increments it. To round it to two decimals I used round(\$sum, 2) which was assigned to \$total which then got echoed. To format the values I used a function called number_format() to format all values to have 2 values after the decimals. When just doing the round I was getting a problem with the 2nd array giving 0.1 only without the 0 after the 1 so I had to format.



Task #3 - Points: 1

Text: Link to Problem 2 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.

Note: This link won't be active until you merge the pull request, so it's expected to not exist at the time of filling this task

URL #1

<https://jhr4-it202-008-prod-af7c79552123.herokuapp.com/PHP-Problems/problem2.php>



Problem 3 (3 pts.)



Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have its values converted to a positive version of the value AND converted back to the original data type.

Requires at least 2 screenshots (code + output from heroku dev)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/>	1	Edits were done only in the bePositive() function and original template code/comments remain

#1	1	untouched
<input type="checkbox"/> #2	1	Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)
<input type="checkbox"/> #3	5	Passed in array's values will get converted to a positive version AND converted back to the original data type
<input type="checkbox"/> #4	1	Includes code comments with student's ucid and date
<input type="checkbox"/> #5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

☐ Large Gallery



Checklist Items (3)

#1 Edits were done only in the bePositive() function and original template code/comments remain untouched

#2 Only \$arr is used (no direct usage of \$a1, \$a2, \$a3, \$a4)

#4 Includes code comments with student's ucid and date



Checklist Items (2)

#3 Passed in array's values will get converted to a positive version AND converted back to the original data type

#5 Output of code includes heroku dev URL with student's ucid visible

Heroku deploy of PHP problem 3.

Code for PHP Problem 3.

Task #2 - Points: 1

Text: Explain your solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Clearly explains how the code/logic solves the problem (mentions both the conversion to positive and conversion to original data type)

Response:

For this question I tried the foreach loop just to test it out instead of the while loop, as I've never used a foreach loop before. After this I implemented the ability to make the numbers all positive. I did this through math by square rooting the product of the number times its self. However, I had to then check for the data type and make it into the same datatype as the original.

I added if statements in the loop that checked what the data type of the numbers in the array are (string, float, or int). Each if statement was a different data type. I checked the data type with a is_type() function/method.

Then, I added the math for each if statement and casted it to the data type that it originally checked.

Then I echoed the number and a var_dump of the number under it to make sure it's the original data type. (this was necessary as when echoing the string it didn't have the quotation marks to make sure it was a string type).



^ COLLAPSE ^

Task #3 - Points: 1

Text: Link to Problem 3 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.

Note: This link won't be active until you merge the pull request, so it's expected to not exist at the time of filling this task

URL #1

<https://jhr4-it202-008-prod-af7c79552123.herokuapp.com/PHP-Problems/problem3.php>



Reflection (1 pt.)

^ COLLAPSE ^

Task #1 - Points: 1

Text: Reflect on your experience

Details:

Talk about any issues you had, how you resolved them, and anything you learned during this process.

Provide concrete details/examples.

Response:

Talk about any issues you had, how you resolved them, and anything you learned during this process.

The one issue I had was in the last problem. I didn't know how to make the output datatype the same as the original datatype via a function. Therefore, I basically brut forced it with the if statements that checked if the number "is_type" for each type (string, int, and float). It would technically not work if there's another datatype in it like an array consisting of an array with the number.


I also had a issue with the second problem however, it was solved by a function on W3schools that let me format the number.

Some things I learned were the foreach loop which made looping over arrays very simple.

^ COLLAPSE ^

Task #2 - Points: 1

Text: Include the pull request link for this branch

 Details:

The correct link will end with /pull/ and a number.